

ASSEMBLY LINE DESIGN: A SURVEY

**Brahim Rekiak*, Alain Delchambre*,
Alexandre Dolgui** and Antoneta Bratcu****

** CAD/CAM Department
Université libre de Bruxelles
Av. F. D. Roosevelt 50, CP 165/14
B -1050 Brussels, Belgium
Email: brk@fabricom.be, adelch@ulb.ac.be*

*** Industrial Engineering Department
University of Technology of Troyes
12, rue Marie Curie B.P. 2060
10010 TROYES Cedex France
Email: dolgui@utt.fr*

Abstract: The problem of optimal design of the assembly lines is considered. The paper is especially focused on the line balancing and resource planning step for the preliminary design stage. A survey of existing methods and software tools are given.
Copyright © 2002 IFAC

Keywords: Manufacturing systems, Assembly, Design, Optimisation.

1. INTRODUCTION

Assembly line are the most commonly used method in mass production environments because they enable assembly of complex products by workers with limited training. The main objective of assembly systems designers is to increase the efficiency of the line by maximising the ratio between throughput and required costs. Thus, assembly line design is a problem of considerable industrial importance.

Each company taking the decision that assembly is a strategic core activity, is faced with the question of how to utilise and develop the assembly system in the most efficient way in order to become successful. The assembly activities performed within the assembly system not only determine the final quality of the products, but also affect time-to-market, delivery, etc. Emphasising the way in which an assembly system is designed will, therefore, control its efficiency and quality. For manual assembly line

the most interesting performance index is stations workload's balancing.

This paper is dedicated to works on the assembly line design, especially the line balancing and the resource planning. It gives a brief description of the methods applied to the assembly line design problems: exact methods, heuristics, metaheuristics, etc.

2. SIMPLE-PRODUCT LINE

Lucertini (Lucertini, 1998) has shown that despite hundreds of works on assembly line design, little number of companies use published techniques to balance their lines. One of the reasons is that the models usually adopted to solve the problem suffer of a substantial loss of information. In fact, little work has been done to model the full range of practical considerations in assembly line design. Most of the time, line balancing methods tackle

linear assembly line, without confluence of or separation into sub-lines. The common performance indices are the cycle time and the number of stations, whereas other factors may also heavily affect the system performances. Some of these, such as traffic problems, station congestion and transportation network are often considered to be marginal in the design stage of the assembly line. An introduction to optimal approaches of the assembly line balancing (ALB) problem and its varieties can be found in (Baybars, 1986) and (Scholl, 1999).

2.1. Bin Packing approach

The ALB problem usually refers to single product assembly line. The simplest version of the problem is the well-known bin packing problem (BPP), where the aim is to pack uni-dimensional 'objects' of various sizes into as few identical 'containers' of a given capacity as possible. Assuming that the durations of the tasks to be performed are fixed, they can be taken for the unique dimension of the 'objects'. Then, casting the cycle time as the capacity of the bins, the problem becomes how to distribute operations among stations in such a way that no station overflows the cycle time and a minimum number of stations is used. Note however that this simple formulation does not take into account the precedence constraints between operations, which amounts to assume that a product can be assembled by any succession of operations. Fitted with acyclic precedence constraints (Sacerdoti, 1977), the BPP becomes the simple assembly line balancing problem (SALB).

Two versions of this problem are known (Scholl, 1999):

SALBP-1 consists to assign tasks to stations so that the number of stations is minimised for a given production rate.

SALBP-2 aims maximising the production rate, or equivalently, minimising the sum of the idle times for a given number of stations.

Wee and Magazine (1982) showed that SALBP-1 is NP-hard since it is a special case of the BPP which is known to be NP-hard. The authors proposed to solve the problem by means of the generalised bin packing heuristics. A B&B method similar to the one used for the BPP is used. The authors also proposed techniques used in the BPP like the first-fit-decreasing (FFD) rule and the immediate-update FFD (IUFFD) to the ALBP. In the FFD rule method, the tasks are listed in a non-increasing order of process time. In the case of IUFFD, the set of available tasks is immediately updated and arranged in non-increasing order of process time. In general, these two heuristics give good results.

2.2. Precedence constraints

Precedence graph generation. An ALB problem can be presented as an optimally partitioning the precedence graph. Works dedicated to the systematic generation of the precedence graphs are not so many. Having conceived LEGA, a powerful software for obtaining the set of all the assembly trees (sequences) for a given product (Henrioud and Bourjault, 1991), researchers from LAB (Besançon, France) intended to give a method of connection between this set and the corresponding precedence graph(s). Thesis of Chen (1996) and of Bratcu (2001) concern the systematic generation of the set of precedence graphs equivalent to a given set of assembly sequences.

More recent works have focused on deducing the precedence graphs in the multiproduct case (for a family of products). The method proposed by Naphade *et al.* (1999a, 1999b), consisting in the generation of all the precedence graphs starting from a given set of precedence constraints – failed in treating the disjunctive constraints, as it could produce cyclic graphs. De Lit (2000) proposed the improvement of this method, by constructing a tree of all possible solutions, by successively adding precedence constraints.

Optimisation. Salveson (1955) was the first to empirically address the ALB problem. He described it as a set of precedence constraints between tasks that must be adhered to. Salveson formulated the SALB problem as a linear programming problem including all possible combination of station assignments. His model, by definition, can result in splitting tasks and may generate infeasible solutions. Bowman (1960) was the first to provide a 'non-divisibility' constraint, by changing the linear programming formulation to a zero-one integer programming. Due to the high number of variables the method requires, even for small problems, the work of both Salveson and Bowman is impractical to find optimal solutions.

The SALBP can be also represented by a *tree* in which each path corresponds to a feasible solution with each arc representing a station. Jackson (1956) was the first to propose an algorithm for SALBP-1 using the notion of a tree. A station is termed maximal if no task can be assigned to it without violating the precedence and the cycle time constraints. The method starts by generating all feasible assignments to the first station (one of these feasible assignments will obviously be part of the optimal solution). Then it generates all feasible assignments to the second station, given the first station assignments. Then for each first-second station combination, all feasible solutions are constructed for the third station, etc.

The process is repeated, each time adding one station. Jackson uses dominance arguments to eliminate inferior feasible assignments: after

generating all feasible assignments, these sets are compared to eliminate subsets that contain exactly the same tasks as other subsets and to eliminate the dominated subsets. A set is said to be dominated if it contains fewer tasks than another subset and does not contain any tasks that are not also in the other subset. Jackson's method does not require the generation of all feasible sequences, but still uses exhaustive search of all remaining possibilities.

Agnietis (1997) addressed the problem of assigning and sequencing tasks to stations with the objective of minimising the inventory costs or the completion time of a set of identical jobs. They consider the standard (tasks and their pre-processing tasks are grouped on the same station) and the look-ahead (tasks and their pre-processing tasks are not necessarily grouped on the same station) assignment policies. Three types of sequencing policies are proposed where the main difference is the order in which the tasks assigned to a station are executed. The method is limited to a single product and it is based on the dynamic programming technique.

Shtub (1990) pointed out the gap existing between literature dealing with the ALB problem and technical documentation used by engineers to describe assembly operations. Comparing the precedence graph to the assembly chart, it is obvious that precedence graph which presents tasks by their times and precedence relations does not fully describes the relations between a task and the subassemblies on which this task is performed. An ALB method based only on precedence graph is likely to generate solutions that minimise idle time by loading each station with tasks performed on several subassemblies. Such solutions are in contradiction with the very basic principles of improving work methods and enriching employee jobs. The author proposed two ways to deal with the relationship between a task and the subassemblies on which the task is performed. The first approach aims to minimise the number of subassemblies handled by each station. The second approach aims to constraint the number of sub-assemblies by each station. The method is applied to SALBP-1 and SALBP-2.

On the evolutionary algorithm side, to our knowledge, the first attempt was by Falkenauer and Delchambre (1992) who used a grouping genetic algorithm (GGA). The authors generalised their bin packing GGA to obtain a fast algorithm supplying high-quality approximated solutions of the ALB problem. One advantage of their method was its ability to handle problems with sparse, even empty precedence constraints, thanks to its bin packing 'ancestor'. Another advantage lies with the fact that the GA is a gradual improvement method, thus giving the user the possibility when the computational resources allow it to extend the search and to continue to improve the currently available solution. Note that the mechanism of complying with

precedence constraints applies equally well to the hybrid GGA of (Falkenauer, 1996).

2.3. Process time

The task process time is an essential parameter in the assembly line balancing. Indeed, depending on the nature of tasks and the skills of operators or the reliability of the dedicated machine or robot, the operation time may vary. Simple tasks may have a small process time variance, while complex and unreliable tasks may have high varying execution times. In the case of human workers, factors such as skills, motivation, communication among the group, etc., have a great influence on the whole assembly line. The following situations may arise.

Deterministic time. In the case of manual assembly line, the task time will stay constant only in case of highly qualified and motivated workers. Modern machines and robots are able to work permanently at a constant speed. The use of deterministic times is always justified if the expected process time variability is sufficiently small (this happens case of simple and well defined tasks).

Pinto *et al.* (1983) discussed processing alternatives in a manual assembly line as an extension of SALB by relaxing the assumption that all stations are identical. Each processing alternative was related to a given set of tasks. The problem was to decide which alternative (from the existing ones) to use in order to shorten the task duration for a given total cost. Since the line is manual, each task may be performed at any station. The authors proposed an integer programming formulation of the problem. Their solution procedure consisted of a B&B algorithm in which a SALB problem is solved in every node of the tree. Therefore, this method may be used only for a small number of possible processing alternatives. This study presented the interesting advantage of working on a set of precedence constraints instead of a fixed sequence.

Stochastic time. In automated flow line-production varying production rates may result from machine breakdowns (Rekiek, 1998). To incorporate the process time variability (manual or automated line), operation time may be modified by adding the stochastic component reflecting the MTBF (mean time before failure) and the MTTR (mean time to repair). When dealing with manual assembly line, the process time variability may result from the task complexity, skills level and motivation of the worker, etc.

Moodie and Young (1965) presented a modified formulation of ALB problem that includes task time variability. Tasks times are considered to be independent normally distributed with known mean and variance. Their standard heuristic places tasks into stations starting with tasks having the longest processing time. A task cannot be placed into a

station unless all of its immediate predecessors have been already assigned. The shortest time algorithm places tasks into stations according to the shortest processing time.

Suresh and Sahu (1994) developed a simulated annealing heuristic for the ALB problem. They considered the problem of stochastic task durations. They only considered single objective problems. Their aim was to minimise the smoothness index introduced in (Moodie and Young, 1965). The smoothness index intends to distribute work among stations as evenly as possible

McMullen and Frazier (1998) presented a 'simulated annealing' (SA) method to address the ALB with multiple objectives. Two kind of objectives were used: the single and the composite objectives. Three single objectives which are: (1) minimise the design cost associated with both labour requirement and equipment requirement, (2) minimise the smoothness index which intended to distribute work into the station as evenly as possible, and (3) minimise the probability of lateness to deal with the stochastic nature of task durations. They also introduced three composite objectives which are expressed as the weighted sum of the first and the third objective. Different weights were attributed to objectives to lay stress on the favourite criteria. The choice of relative weights used for the different composite objectives was quite arbitrary—the approach was very over-fitted. The authors pointed out the difficulty to deal with the multiple objective nature of the problem using the weighted sum approach.

Suresh (1996) used a GA to solve the single model stochastic version of ALBP. A modified GA, working with two populations (one allowing infeasible solutions), and exchange of specimens at regular intervals, is proposed for handling irregular search spaces, i.e. the unfeasibility problem due to precedence relations. The authors claimed that a population of feasible solutions would lead to a fragmented search space, thus increasing probability of getting trapped in a local minimum. Infeasible solutions can be allowed in the population only if the genetic operators can lead to feasible solutions from infeasible ones. Some solutions are exchanged at regular intervals between the two populations; the exchanged solutions have the same rank of fitness value in their own populations. The results of the experiments indicated that the GA working with two populations can give better results than the GA with only feasible population.

Sotskov and Dolgui (2001) address the SALB-1 problem. Subset \tilde{V} of set V of all operations includes manual operations, for which it is hard or even impossible to fix processing time for the whole life cycle of the assembly line. It is assumed that if $j \in \tilde{V}$, then given operation time t_j can be different for different cycles of production process. They introduced the notions of stability ball and stability

radius of optimal line balance with respect to perturbations of some operation times. It is proven the necessary and sufficient conditions for stability of an optimal line balance. The authors present upper bound of stability radius of an optimal line balance. It should be noted that all these conditions may be tested in polynomial time, which is important for real assembly lines with a large number of operations.

Hidden times. In the case of automated stations, the global duration of a 'complex' operation is not always the sum of the operating times of every equipment in the group. A hidden time represents parts of the operation that can be executed at the same time than other tasks belonging to the same station (Pellichero, 1999).

Minzu and Henrioud (1997) presented a general algorithm for the stations determination, for mono-variant products. It was based on the use of a so-called assembly graph, which combines the advantages of the assembly tree and the precedence graph, i.e. it combines the precedence constraints information with geometrical information like the relative orientations of the components. A method of tasks-to-equipment assignment based on this graph was proposed, as well as the operative sequences for the equipment and the computation of the process-time of a station. These methods are very interesting, especially for the evaluation of the process time of a station which allows taking into account the hidden times, the operators moving, and the setup time. The authors used the 'constraint propagation' method embedded in the 'Prolog' language and a backtracking approach. A depth-first 'backtracking' method was used during the tasks-to-station phase. However, two drawbacks can be mentioned for these methods. The first one was that they did not take cost and availability factors into account. The second drawback of these methods, already mentioned in (Petit, 1999), was that they had not been linked to an equipment database and were not really considering the technical station design.

Dynamic time. In case of human workers, systematic reductions are possible due to learning effects or successive improvements of the production process. Thus the manual lines have to be balanced on average, the accent must not be highly put on the process time. The equal piles philosophy is well suited for such lines (Rekiek *et al.*, 1999, Rekiek 2000).

Bartholdi and Eisenstein (1996) introduced the concept of 'operation of bucket brigades' for manual line, used for example in the automotive industry. It works as follows: each worker carries a task towards completion; when the last worker finishes his product he sends it off and then walks back upstream to take over the work of his predecessor, who walks back and takes over the work of his predecessor and so on. In such a configuration the workers play the role of the buffers, by supporting the variability of the

operating times (due to stochastic phenomena or differences between the operations according to the variants).

The main advantage of this kind of lines is the self-balancing: they can be configured to spontaneously maintain the optimal production rate. It is sufficient to place the workers from slowest to fastest in the sense of products advancing. A simulation study of such lines is reported by Bratcu and Mînză (1999). The same authors have shown that bucket brigades can be regarded as hybrid dynamic systems (with autonomous switching and autonomous jumps), whose stability properties are related to the property of self-balancing (Bratcu, 2001; Bratcu and Mînză, 2001).

2.4. Equipment selection

Most of the research on balancing deals with the so-called SALB problems in which no alternative equipments are considered. That is, each task's process time is fixed, and the aim is to determine the sets of tasks to be performed on each station. With respect to more than one type of equipment, there are relatively few studies that address this problem. The problem is known as the Resource planning (RP). The SALB problem is proven to be a NP-Hard problem, by the way the RP problem is NP-Hard as well.

Graves and Withney (1979), in one of the first works on the field, presented a method for the single product equipment selection problem. The approach was based on linear programming techniques and solved by a B&B procedure. The method did not deal with precedence constraints, rather a sequence, and permitted non-serial line layouts in which an assembly unit visits more than once a given station. The goal was to select equipment and make task-assignment so as to minimise the sum of fixed and variable costs.

Graves and Lamar (1983) extended the work of Graves and Withney (1979) and developed an integer programming cost-based model to the automated system design problem. A column generation procedure was applied to solve the integer formulation of the problem. Their aim was to determine the type and the number of stations, and the operations assigned to these stations. The approach used a fixed assembly sequence, which is a first inconvenience, but their main limitation, as in (Graves and Withney, 1979), was that they permitted non-serial line-layouts. Their model did not explicitly try to balance the total workload across the stations, rather the design criterion is to minimise total system costs, while satisfying a desired production rate. The station loads were often highly unbalanced. As a result of allowing unrestricted floor layouts for the problem, the solutions found were not necessarily physically feasible.

Gustavson (1986) developed heuristic methods for solving both the single and multiple product equipment selection problem. The author proposed an alternative to avoid the problem of the non-serial line layouts. These heuristics were also based on a fixed assembly sequence with the inconveniences already mentioned above. Of course, being heuristics, these methods cannot guarantee that an optimal solution will be found.

Graves and Holmes (1988) considered the design problem with several equipment alternatives, when multi-products are assembled on the same line. They assumed a complete ordering of tasks of the same product and large similarities among different products. These assumptions resulted in a relatively small number of candidate stations and therefore simplified the problem considerably. The method seeks to assign tasks to stations and selects equipments for each of them. It aims to minimise the total cost of the assembly line and it is composed of two steps. The first step consists in enumerating all candidate stations for the system and selecting the least-cost resource type for each candidate station. The second step consists in finding the least-cost assembly system. For this purpose, a graph in which each candidate station corresponds to an arc is used, the length of the arc being proportional to the cost of the station. The least-cost assembly system is then found by solving a shortest-path problem in the network of feasible stations. They addressed this problem, using a method similar to the one of (Gutjahr, 1964). Their algorithm enumerates all feasible stations, selects the best equipment for each, and then chooses the best set of stations. The method guarantees the feasibility of the layout by restricting the system to a linear floor layout. However, given the implicit enumeration of all possible stations, the method is impractical for large problem instances.

Falkenauer (1995) proposed a genetic algorithm for ALB with 'resource dependant tasks times' algorithm based on a GGA and a B&B algorithm. The method was able to supply a well balanced and cheap assembly line. The minimal and maximal number of stations was computed by solving the classical ALB problem with respectively the fastest and slower resource assigned to all tasks. The GGA distributed the tasks onto stations, while the B&B algorithm selected the optimal resource for each station.

Sysoev and Dolgui (1999) present an equipment selection problem as a multi-criteria decision making problem. To solve it, the authors propose an iterative Pareto optimization method with heuristic procedures of selection at each step. The convergence of this method is proved and an application example is given.

Bukchin and Tzur (2000) developed an optimal method and a heuristic algorithm to design flexible assembly line when several equipment alternatives

are available. Tasks are subject to precedence constraints. The objective was to minimise total equipment cost, given a pre-determined cycle time. They developed an exact B&B algorithm which was capable of solving practical problems. The algorithm's efficiency was enhanced due to the development of good lower bounds, as well as the use of dominance rules to reduce the size of the tree. They also suggested to use a B&B based heuristic procedure for large problems. For large problems that cannot be solved by the optimal algorithm, the authors developed a heuristic procedure. The procedure's control parameters may be chosen according to the size of the problem. The control parameter determines how many nodes of the tree may be skipped, and therefore was responsible for the running time, for the memory requirements, as well as for the distance from optimality of the resulting solution. The method dealt with many realistic features of assembly line, unfortunately it did not deal with the operating mode of tasks which is one of the hardest constraints of the problem.

2.5. Additional constraints

The classical line balancing problem can be improved to include other factors in order to deal with real-world problems. Dealing with constraints and user's preferences reduces the number of task-station combination and at the same time the search space of the problem. Such constraints have the same influence on the complexity of the problem as the precedence constraints of the product.

In spite of the existence of many commercial software of 'balancing', less than 10% of companies use such software. The reason is that industrial balancing problems present most of the time, particularities which are not always considered by the standard software. Lapiere (1999) presented the COMSOAL algorithm (Arcus, 1965) programmed on the software package Microsoft Access'97. The author modified the COMSOAL algorithm in order to deal with constraints such as the position (rear, front, centre, etc.) and the level (high and low) of tasks. Thus, the method aims to avoid grouping tasks having different levels on the same station. Tasks requiring heavy equipment or parts must not be assigned to the same station. The approach is very interesting, however, we think that the ALB problem is quite difficult to be solved by a simple 'spreadsheet' software. We believe that the simple version of the ALB is quite difficult to be solved by such a method. Anyway, the author recommends using metaheuristics to deal with ALB problem.

Operators. Although different variations of the ALB problem have been explored by researchers (Scholl, 1999), little concern has been given to the physical demands placed on workers when assigning tasks to stations. Carnahan *et al.* (1999) proposed a methodology for the ALB considering both

production objectives (cycle time and number of station) as well as worker physical constraints. The method takes into account the physical requirements of tasks. The method is based on the GA and applied to the SALBP-2 where the aim is to minimise the cycle time for a given number of stations. The authors used three search algorithms: (1) a multiple rank heuristic (MRH), (2) a combinatorial GA, and (3) a problem space GA (PSGA). The MRH is a combination of 81 separate heuristics that utilises three search methods (lower bound, upper bound and binary search), three ranking criteria (rank positional weight, number of successors, number of paths), three tasks assignment (forward, backward, bi-directional) and three weighting factors. The sequence-oriented coding combinatorial GA assigns tasks to stations as the classical GA, the main difference is the fact that the number of stations is fixed. Thus, the cycle time constraint is not enforced when assigning tasks to the last station. This ensures that all tasks will be assigned. Tests let the authors to conclude that the PSGA perform better than the MRH and the classical GA.

Buffers. Malakooti and Kumar (1996) used a multi-criteria decision-aid method for ALB problems where the objectives were the number of stations, the cycle time, the buffer size and the total cost of the operations. For the single objective ALB problem with buffers, the authors used a three steps interactive method to minimise the total cost of the line. The method first balanced the line using the RPW algorithm (Helgeson, 1961). It then looked for the required buffer size using an empirical formulation. Finally the total cost of the line is estimated. For the multiple objective ALB problem, the authors used an additive utility function based on decision maker's preferences (weights). They divided the whole into three simple problems: (1) minimise the number of stations when the cycle time is given, (2) minimise the cycle time when the number of stations is given and (3) minimise the total cost when the cycle time is given and the number of stations has to be determined. Solving the three problems for different values of the cycle time and the number of stations may lead to different solutions. The best solution from the efficient alternatives was chosen using a 'ranking alternatives by strength of preferences' method. The attributes of the additive utility function were the cycle time, the number of stations, the buffer size and the total cost.

Dolgui *et al.*, (2002b) consider a buffer allocation problem under machines breakdowns. The line balancing problem is supposed resolved. So, the buffer allocation problem consists in determining the buffer capacities with respect to a given optimality criterion, which depends on the average production rate of the line, the buffer acquisition and installation cost, the inventory cost. A genetic algorithm where the tentative solutions are evaluated with a method based on the Markov models is proposed.

3. GENERALIZED PROBLEMS

3.1. Heuristic models

Single product line. Since the ALB problem falls into the NP-hard class of combinatorial optimisation problems, numerous research efforts have been directed towards the development of computer efficient approximation or heuristics algorithms (Ghosh, 1989).

One of the first proposed heuristic was the ranked positional weight (RPW) (Helgeson, 1961). The main idea behind the RPW heuristic is to first assign the tasks which have long chains of succeeding tasks. The length of the chain can be measured either by the number of successor tasks, or by the sum of the task times of the successor tasks. In the RPW method, the sum of the task process time and the process times of the successor tasks is defined as the positional weight of the task. The tasks are ranked in descending order of the positional weights with ties broken arbitrarily. The tasks are then picked in their ranked order and assigned to stations if (1) all predecessors of the task have already been assigned and (2) the task fits in the remaining time on the station. If a task does not fit in the remaining time on a station it is skipped and the next task in the ranked order is selected. If no task fits into the station, the station is closed and a new station is opened. The tasks are then scanned from the beginning of the list and an attempt is made to assign unassigned tasks to the new station. The process is repeated until all tasks are assigned. Since the weight of an element depends on the process time of its predecessors, the algorithm tends to treat the elements near the end of precedence graph first. This procedure is quite simple, it can deal with user's preferences but gives good results only for simple problems.

The 'reversed ranked positional weight' (RRPW) is the RPW method applied to the problem with the reversed precedence constraints. After a balance is found for the reversed problem, the problem is 'un-reversed' to get a balance solution for the original problem.

Kilbridge and Wester (1961) proposed a heuristic based on the cumulative performance times. First the 'layers' are identified in the precedence graph. Tasks with no predecessors are in the first layer. Tasks that are preceded directly by tasks in layer i are placed in layer $i+1$, etc. Then, it computes the time for each layer (sum of process time of tasks belonging to a given layer), as well as the cumulative performance time of each layer (sum of its time and times of the layers it succeeds). Moving from the left to the right in units of a layer, it finds the layer that just fills or overfills the station. It begins by the first layer and go in order through the remaining layers. Tasks of a given layer are assigned to a given station unless the precedence constraints will be violated or the cycle time will be exceeded, otherwise they are assigned to the next station and so on. The method is a search

algorithm which is restricted to looking one layer behind and one layer ahead in filling the stations. This heuristic is easy to implement and gives good solutions for dense and uniform graphs. The more the graph contains sprays, less the method yields good balancing.

Multiple product line (mixed/batch production). Traditionally assembly line balancing and variants ordering (for mixed-production) have been considered as two separate, but related problems. Most of the research on assembly line sequencing considers scheduling products after assembly line balancing. By separating the two problems, sub-optimal solutions are often obtained. Rekiek and Delchambre (1998) introduce the balance for ordering (BFO) concept to treat both problems simultaneously.

In (McMullen and Frazier, 1998) the trade, transfer, compression as well as expansion heuristics are used as local improvements. The initial solution is based upon the 'incremental utilisation' heuristic (Gaither, 1996).

Lee and Johnson (1991) proposed an iterative method based on integer programming, depth-first B&B and queuing network analysis. The approach allowed to deal with multiproduct assembly line. The objective was to minimise the cost of work-in-process inventory, machine investment and maintenance and material handling. They considered five design factors: the number of stations, the number of parallel machines, the tasks assignment, the number of pallets and fixtures and finally the number of automated guided vehicles. The method suffered from a severe restriction: each station consists in a single machine or in identical parallel machines.

3.2. Line configurations

Several line configurations are possible, for example:

Serial line: single stations are arranged in a straight line along a conveyor belt. Each station performs one or more tasks on the partially finished product (Baybars, 1986; Scholl 1999).

U-Line: as a consequence of introducing the just-in-time production principle it has been recognised that arranging the stations in a U-line has several advantages over the traditional configuration. Thus, workers acquire multiple skills leading to higher motivation, improve quality of products and increased flexibility (Scholl, 1999).

Parallel stations: with high production rates, the longest task time sometimes exceeds the specified cycle time. A common remedy is to create stations with parallel or serial posts, where two or more workers perform an identical set of tasks. This procedure reduces the average value of the task

duration proportionally to the number of workers on the station (McMuller and Frazier, 1998).

Bard (1989) presented a dynamic programming algorithm for solving the ALB problem with parallel stations. Solutions represent a trade-off between the minimum number of stations required to achieve a balance and the cost of installing additional facilities. Both equipment cost and task cost are considered. The algorithm takes into account the unproductive time during a cycle.

Lee *et al.* (1999) presented a GA to deal with ALB problem with parallel stations. The GA is hybridised with two local search procedures: the trade and transfer heuristics. The trade procedure is used to exchange tasks between two adjacent stations. The transfer procedure has two varieties: the expansion transfer aims to create additional stations if needed, while the compression transfer aims to decrease the number of stations. The method aims to minimise the weighted sum of (1) the total cost of the line and (2) the lateness across all stations. They used the sequence-oriented representation—the tasks are sequentially listed in the order in which they are assigned to the stations. This representation is clearly not grouping-oriented and leads to an inefficient encoding.

Parallel lines: permits to decrease the failure sensitivity and increase the flexibility of the system. The use of parallel lines allows the enlargement of the cycle time(s). Thus, the extent of labour division is relatively small so that only few operators are employed in every parallel line. These units are organized as autonomous work teams.

Süer (1998) proposed a 3-phase methodology to deal with assembly lines having high production volume. The objective is to determine the number of assembly lines (in parallel) with minimum total manpower. In the assembly line balancing phase, tasks are grouped into stations for various configurations (1-station, 2-station, ..., n-station; where n is the number of tasks). Thereafter, the parallel station phase determines the number of operators that should be assigned to each station (parallel stations). The parallel lines phase determines the number of lines and their configuration with the objective of minimising the total number of operators.

Workcenters: for complex products, the assembly system is most of the time decomposed into subsystems which are easier to manage than the entire one. The line is decomposed into several linked sub-lines (workcenters). The routing of a product from one workcenter to another is fixed, according to a line flow topology. The main topology of the line is not necessarily a linear one (Rekiek *et al.*, 1999).

Parallel tasks (blocks). Dolgui *et al.* (1999, 2000a, 2001, 2002a) consider a new problem of line

balancing for paced transfer lines with workstations in series and with block of parallel operations at the workstations. The problem is to minimise a cost associated with the number of workstations and the number of blocks, providing a given cycle time and specified technological constraints. Two methods are developed to solve the problem. The first method is based on a transformation of the initial problem into a constrained shortest path problem (Dolgui *et al.*, 1999, 2000a). The second method uses mixed integer programming formulations of the problem (Dolgui *et al.*, 2000a, 2001, 2002a). Both graph and mixed programming methods are combined with some heuristic techniques in order to solve large-scale problems (Dolgui *et al.*, 2000a).

Robotic assembly lines. Rubinovitz and Bukchin (1993) introduced the robotic assembly line balancing (RALB) algorithm for solving single model ALB problems. The method was based on a B&B algorithm and aimed to design (balance) robotic assembly line when several robot types are available. Their model supposed that all the equipment alternatives have an identical purchasing cost. The objective was to find a line balance which minimises the number of stations for a given production rate. Rubinovitz and Levitin (1995) proposed a genetic algorithm for this problem.

3.3. Design objectives

Depending on the line layout many objectives may be sought after.

For SALB, Minzu and Henrioud (1997) proposed a 'kangaroo' algorithm (a stochastic descent method) to treat the problem of assembly line with a fixed number of stations. The stochastic descent method aims to minimise the maximum work content of the stations, which leads to a well balanced line. The computational tests proved that the 'kangaroo' algorithm supplies good solutions in a small number of iterations.

EPLBP: (Equal Piles for Assembly Line Balancing Problem): warrants to obtain the desired number of stations, and tries to equalise the station loads (Rekiek *et al.*, 1999).

Rachamadugu (Rachamadugu, 1991) investigated the levelling of workloads across stations which is an important problem especially in manual assembly lines. Indeed, uneven assignments are viewed as inherently unfair and usually call for some kind of compensatory actions from management such as differential pay. A mean absolute deviation of workloads criteria is used as an objective function to distribute the workload as evenly as possible. The author identified the notion of dominance between solutions and incorporated it into an iterative heuristic. Given any heuristic or optimal solution to a SALBP-1 or SALBP-2, the levelling procedure

reassigns tasks among the stations to smooth the workload across stations. It is a kind of local optimisation used after the global one. The proposed method yields better allocation of work in the case of SALBP-1 and SALBP-2.

SALBP-G: aims to minimise the sum of the idle times with a varying production rate and a changing number of stations (Scholl, 1999).

RP: find an assignment of the tasks to stations along the line, and determine the resources to be allocated to each task among the possible ones for that task (Rekiek et al., 1999). Many objectives can be found in the literature: minimise total line cost, maximise reliability, etc.

MOALBP: (multi-objective line balancing problem): most real world problems involve multiple objectives. In the assembly line problem, designers deal with objectives like: line efficiency, smoothness index (imbalance), cost, reliability, buffers, stations space, etc. (Chow, 1990).

Kim et al. (Kim, 1996) developed a GA to solve multiple objective ALB problem. They addressed several types of ALB problem. They considered the following objectives: (1) minimise number of stations; (2) minimise cycle time; (3) maximise workload smoothness; (4) maximise work relatedness (interrelated tasks are allowed to the same station as much as possible); and (5) a multiple objective with (3) and (4). The authors used a sequence-oriented coding, as well as a repair routine to deal with infeasible solutions. A set of ordering-oriented crossover and mutation operators were used. Their emphasis is placed on seeking a set of diverse Pareto optimal solutions. The multiple objective GAs seem to be very promising, in contrast the encoding scheme is not well suited for the grouping problem they deal with.

Lee and Stecke (1996) presented an integrated design support method for flexible assembly systems. The utility of a multi-criteria optimisation tool is underlined in that work. But one of its main weaknesses is that the optimisation was applied only to the cost while satisfying a set of constraints.

Ponnambalam *et al.* (1998) used the sequence-oriented representation for a multiobjective GA (MO-GA). They used 14 simple heuristics (Talbot *et al.*, 1986) to initialise the population and used classical genetic operators to evolve solutions. The method aims maximising the line efficiency as well as the smoothness index. During the execution of the MO-GA, a tentative set of Pareto optimal solutions are stored and updated at each generation.

In the work (Dolgui *et al.*, 2000b) the goal function is to minimise the line life cycle cost per piece. The model is formulated in terms of mixed (discrete and non-linear) programming. For solving the

optimisation problem, a special decomposition scheme is proposed, which is based on parametric decomposition technique.

3.4. Simulation and Interactivity

On the side of interactive and iterative methods, Nevins and Withney (1989) presented a method based on technical and economical considerations. The method helps to construct the cheapest technically feasible assembly line. The method presented the advantage of being complete, but could become very time consuming to apply if the number of operations and the set of usable resources for each operation become too important. Another weakness was that the assembly sequence was fixed before the application of the method. This could be too restrictive and lead the system to miss the most cost-effective solutions.

Praça (1999) proposed an architecture based on multi-agent simulation (called SimBa). The system helps in balancing and distributing the human resources in manual assembly line. The system is composed of a balancing and simulation modules. The balancing module which implements three heuristic rules from (Helgeson, 1961) and (Boctor, 1995) allows the user to obtain different line configurations. The simulation module is used to evaluate the performance of the proposed configurations. The aim of agent design is to create programs which interact intelligently with their environment. The multi-agent system is introduced to simulate the human behaviour in a given assembly line.

Tsai and Yao (1993) proposed an integer programming model combined with a simulation adjustment phase. The approach was used to design a flexible robotic assembly line which produces a family of products. Given the work to be done at each station, the demand of each product and a budget constraint, the method determined the robot type and number of robots required in each station along a serial robotic line. Their objective was to minimise the standard deviation of the output rates of all stations, which is the measure for a balanced line.

Okano (1993) proposed a generator of logical expressions able to generate all feasible resource groupings for a given sequence of assembly operations. After obtaining valid combinations of resources, a simulation tool was used to evaluate the different solutions and select the best one. Since the method was based on an assembly sequence, a first fit heuristic method was used to group tasks, where the constraint was the cycle time calculated on the basis of production volume. The criteria for the grouping of operations and resource assignment were the production volume and the production cost. The approach was based on the risky assumption that the

constraints of the problem will avoid the combinatorial explosion of the possible solutions set.

3.5. Software tools

A lot of software development companies try to offer the best optimisation solution for the assembly systems design. They are confronted with a very sharp concurrency fight, giving a high dynamics to the concerned market. Starting from some basic design conception – such as 3D product modelling, multiple users, web accessible data bases, friendly graphical interfaces and, not at least, powerful interactive simulation tools – and being fully object-oriented, the commercial software products offer a virtual environment to assembly systems design and management.

Delmia (see <http://www.delmia.com>) from Dassault Systems company presents ErgoFab, a modular tool aiming at covering all the production optimisation aspects: line balancing, cost optimisation, resource planning, etc. Offering three user options – manually balancing, half automatic and fully automatic balancing – ErgoFab offers also simulation facilities, by either Simple++, or Dosimis-3, or Arena based integrated packages. Line balancing is started right from reading the production structure, yielding the work plans and imposing the precedence constraints. However, this software has limited capabilities of working with hidden times of operations.

Some of the products use classical, well known modelling concepts and optimisation algorithms – for example, Tecnomatix (<http://www.tecnomatix.com>) gives sequencing and line balancing optimal solutions based upon COMSOAL. Whereas others try to minimise the skill level required to the user: Simas II (<http://www.cimpact.ch>) works with user defined assembly rules; Streamline of Factory Logic (<http://www.factorylogic.com>) uses universal machine models. Use of recent modelling techniques it can be noticed also as a trend: knowledge data base based modelling (KAPES) – allowing a more accurate representation of experts experience – or either trials to model the human behaviour (CircuitCam, <http://www.contax.co.uk>). There are software packages developing an iterative design (and redesign) approach, while minimising the number of iterations until finding a good (and possibly optimal) solution (EASE, OptiLine, etc.).

4. CONCLUSION

Despite the number of works on assembly line design and balancing, academic algorithms are not used by industrial companies. This is because, despite their effectiveness and the easiness of their use, they use little data and suffer on substantial loss of information, solving fictitious rather than industrial problems. As many practical optimisation problems,

the assembly line design must be formulated as a multi-objective problem rather than to aim at minimising the number of stations or the imbalance between the stations. Efficient methods in Assembly Line Balancing (ALB) and Resource Planning (RP) should be able to deal with conflicting objectives or take users preferences into account. They should be quick enough to allow the designer testing many alternatives.

REFERENCES

- Agnetis, A. and C. Arbib (1997). Concurrent operations assignment and sequencing of particular assembly problems in flow lines. *Annals of Operations Research*, **69**, 1-31.
- Arcus, A.L. (1966). COMSOAL: A computer method of sequencing operations for assembly lines. *International Journal of Production Research*, **4**, 259-277.
- Bard, J.F. (1989) Assembly line balancing with parallel workstations and dead time, *International Journal of Production Research*, **27**, 1005-1018.
- Bartholdi J.J., and D.D. Eisentein, (1996). A production line that balances itself, *Operations Research*, **44**, 21-35.
- Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing, *Management Science*, **32**, 909-932.
- Boctor, F.F. (1995). A multiple-rule heuristic for assembly line balancing, *J. of Oper. Res. Society*, **46** (1), 62-69.
- Bowman, E.H. (1960). Assembly line balancing by linear programming, *Operations Research*, **8** (3), 385-389.
- Bratcu, A. and V. Mînză (1999). A Simulation Study of Self-balancing Production Lines. In: *Proceedings of the 1999 IEEE International Conference on Intelligent Engineering Systems (INES '99)*, pp. 165-170, Stará Lesná, Slovakia.
- Bratcu, A and V. Mînză (2001). Modeling and Analysis of Self-balancing Manufacturing Lines as Hybrid Dynamic Systems. In: *Preprints of the 9th IFAC/IFORS/IMACS/IFIP Symposium on Large Scale Systems: Theory and Applications (LSS 2001)*, pp. 6-11, Bucharest, Romania.
- Bratcu, A. (2001). *Détermination systématique des graphes de précédence et équilibrage des lignes d'assemblage*, Ph.D. Thesis, Université de Franche-Comté, Besançon, France.
- Bukchin, J. and M. Tzur (2000). Design of flexible assembly line to minimize equipment cost, *IIE Transactions*, **32**, 585-598.
- Carnahan, B.J., M.S. Redfen and B.A. Norman (1999). *Incorporating physical demand criteria into assembly line balancing*, Pittsburgh University, Internal Report TR99-8.
- Chen, K. (1996). *Contribution à une méthode systématique de détermination des graphes de précédence pour l'assemblage*, Ph.D. Thesis, Université de Franche-Comté, France.

- Chow, W.-M. (1990). *Assembly line design - methodology and applications*, Marcel Dekker, New York.
- De Lit, P. (2000). *A comprehensive and integrated approach for the design of a product family and its assembly system*, Ph.D. Thesis, Université Libre de Bruxelles, Belgium.
- Dolgui, A. (2001). Automated Production Line Design (Preface for the Special Track), *Management and Control of Production and Logistics: A Proceedings Volume*, Pergamon, Elsevier Science.
- Dolgui, A., N. Guschinski, and G. Levin (1999). On Problem of Optimal Design of Transfer Lines with Parallel and Sequential Operations, In: *Proceedings of the 7th IEEE International Conference ETFA'99*, Barcelona, vol. 1, pp. 329-334, IEEE Press.
- Dolgui, A., N. Guschinsky and G. Levin, (2000a). *Approaches to balancing of transfer line with block of parallel operations*, Preprint N°8, 42 pages. University of Technology of Troyes/ Institute of Engineering Cybernetics, Minsk.
- Dolgui, A., N. Guschinski and G. Levin (2000b) Optimizing Parameters of Transfer Lines with Parallel Machining, In: *Proceedings of the 16th International Conference on CAD/CAM, Robotics and Factories of the Future*, Trinidad W.I., pp. 317-324. IEE/ISPE.
- Dolgui, A., N. Guschinski, and G. Levin (2001). 'A Mixed Integer Program for Balancing of Transfer Line with Grouped Operations', In: *Proceedings of the 28th International Conference on Computer and Industrial Engineering*, Florida, USA. pp. 541-547.
- Dolgui, A., N. Guschinski, Y. Harrath and G. Levin, (2002a). Une approche de programmation linéaire pour la conception des lignes de transfert, *European Journal of Automated Systems (RAIRO-APII-JESA)*, **36**, to appear
- Dolgui, A., A. Ereemev, A. Kolokolov, and V. Sigaev, (2002b). A Genetic Algorithm for Allocation of Buffer Storage Capacities in Production Line with Unreliable Machines. *Mathematical Modelling and Algorithms*, to appear.
- Falkenauer, E. (1996). A hybrid grouping genetic algorithm for bin packing. *Journal of Heuristics*, **2** (1), 5-30.
- Falkenauer, E. (1995). Solving equal piles with a grouping genetic algorithm, In: *Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA95)*, pp. 492-497. Morgan Kaufmann Publishers, San Francisco.
- Falkenauer, E. and A. Delchambre (1992). A genetic algorithm for bin packing and line balancing, In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1186-1192, IEEE Computer Society Press.
- Gaither, N. (1996). *Production and operations management*, Belmont, CA: Duxbury.
- Ghosh, S. and R. J. Gagnon (1989). A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems, *Int. J. of Prod. Res.*, **27**, 637-670.
- Graves, S. C. and D. E. Withney (1979). A mathematical programming procedure for equipment selection and system evaluation in programmable assembly, In: *Proceedings of the IEEE Decision and Control*, pp. 531-536. IEEE Press.
- Graves, S. C. and B. W. Lamar (1983). An integer programming procedure for assembly system design problems, *Operations Research*, **31**(3), 522-545.
- Graves, S.C. and C. Holmes Redfield (1988). Equipment selection and task assignment for multi-product assembly system design, *International Journal of Flexible Manufacturing Systems*, **1**, 31-50.
- Gustavson, R. E. (1986). *Design of cost effective assembly systems*, C.S. Draper Laboratory Report, N°P-2661, Cambridge.
- Gutjahr, A.L. and N.K. Nemhauser (1964). An algorithm for the line balancing problem, *Management Science*, **11** (2), 308-315.
- Helgeson, W. B. and D. P. Birnie (1961). Assembly line balancing using the ranked positional weight technique, *J. Ind. Engng*, **12** (6), 394-398.
- Henrioud, J.-M. and A. Bourjault (1991). LEGA: A computer-aided generator of assembly plans. In: *Computer-Aided Mechanical Assembly Planning*, Homem de Mello and S. Lee (Ed), Chapter 8, Kluwer Academic, 1991.
- Jackson, J.R. (1956). A computing procedure for the line balancing problem. *Management Science*, **2**, 261-271.
- Kilbridge, M.D. and L. Wester (1961). A heuristic method for assembly line balancing. *Journal of Industrial Engineering*, **12**, 292-298.
- Kim, Y.K., Kim Y.J. and Y. Kim (1996). Genetic algorithms for assembly line balancing with various objectives. *Computers & Industrial Engineering*, **30** (3), 397-409.
- Lapierre, S.D. and A. Ruiz (1999). Equilibrer une chaîne d'assemblage avec Microsoft ACCESS97. In: *Proceedings of 3rd International Industrial Engineering Conference*, pp. 357-364. Presses Internationales Polytechnique, Montreal.
- Lee, C.Y., M. Gen and Y. Tsujimura (1999). Multicriteria assembly line balancing problem with parallel workstations using hybrid Gas. In: *Proceedings of EDA'99*, pp. 115-122.
- Lee, H.F. and R.V. Johnson (1991). A line balancing strategy for designing flexible assembly systems. *Int. J. of Flexible Manuf. Systems*, **3**, 91-120.
- Lee, H.F. and K.E. Stecke (1996). An integrated design support method for flexible assembly systems, *J. of Manuf. Systems*, **15** (1), 13-32.
- Lucertini, M., D. Paccarelli, and A. Pacifici (1998). Modeling an assembly line for configuration and flow management. *Computer Integrated Manufacturing Systems*, **11** (1-2), 15-24.

- Malakooti, B. and A. Kumar (1996). A knowledge-based system for solving multi-objective assembly line balancing problem, *Int. J. Prod. Res.*, **34** (9), 2533-2552.
- McMullen, P.R. and G.V. Frazier (1998). Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel stations. *International Journal of Production Research*, **36**, 2717-2741.
- Minzu, V. and J-M. Henrioud (1997). Assignment stochastic algorithm in multi-product assembly lines, In: *Proceedings of ISATP'97*, pp. 109-114, IEEE Press.
- Moodie, C. L. and H. H. Young (1965). A heuristic method of assembly line balancing for assumptions of constant or variable work element times. *J. Ind. Eng.*, **16** (1), 23-29.
- Naphade, K. S., R. H. Storer and S. D. Wu (1999a). *Graph theoretic generation of assembly plans, Part i: Correct generation of precedence graphs*, IMSE Technical Report 99T-003, Lehigh University, Bethlehem, Pennsylvania, U.S.A.
- Naphade, K. S., S. D. Wu and R. H. Storer (1999b). *Graph theoretic generation of assembly plans, Part ii: Problem decomposition and optimization algorithms*, IMSE Technical Report 99T-004, Lehigh University, Pennsylvania, U.S.A.
- Nevins, J.L. and D. E. Whitney (1989). *Concurrent design of product and process*, Mc Graw-Hill, New-York.
- Okano, A. (1993). Computer-aided assembly process planning with resource assignment. In: *Proceedings of the IEEE Conference on Robotic and Automation*, Atlanta-Georgia, pp. 301-306.
- Pinto, P. A., D.G. Dannenbring and B. M. Khumawala (1983). Assembly line balancing with processing alternatives: an application. *Management Science*, **29**, 817-830.
- Praça, I.C. and C. Ramos (1999). Multi-agent simulation for balancing of assembly lines. In: *Proceedings of ISATP'99*, pp. 459-464.
- Pellichero, F. (1999). *Computer-aided choice of assembly methods and selection of equipment in production line design*, Ph.D. Thesis, Université libre de Bruxelles.
- Peterson, C. (1993). A tabu search procedure for the simple assembly line balancing problem. In: *Proceedings of the Decision Science Institute Conference*, pp. 1502-1504. Washington, DC.
- Petit, F. (1999). *Interactive design of a product and its assembly system*, Ph.D. Thesis, Université Catholique de Louvain.
- Ponnambalam, S.G., P. Aravindan and G. Mogileeswar Naidu (1998). Assembly line balancing using multi-objective genetic algorithm. In: *Proceedings of CARS&FOF'98*, pp. 222-230. Coimbatore, India.
- Rachamadugu, R. and B. Talbot (1991). Improving the equality of workload assignments in assembly lines. *Int. J. Prod. Res.*, **29** (3), 619-633.
- Rekiek, B.B. and A. Delchambre (1998). Ordering varinbats and simulation in multiproduct assembly lines. In: *Proceedings of VR-Mech'98*, pp. 49-54. Brussels, Belgium.
- Rekiek, B., P. De Lit, F. Pellichero, F. Falkenauer and A. Delchambre (1999). Applying the equal piles problem to balance assembly lines. In: *Proceedings of ISATP'99*, pp. 339-404. Porto, Portugal.
- Rekiek, B. (2000). *Assembly line design*, Ph.D. Thesis, Université libre de Bruxelles.
- Rubinovitz, J. and J. Bukchin (1993). RALB - A Heuristic Algorithm for Design and Balancing of Robotic Assembly Lines. *Annals of the CIRP*, **42** (1), 497-500.
- Rubinovitz, J. and G. Levitin (1995). Genetic Algorithms for Assembly Line Balancing, *International Journal of Production Economics*, **41**, 343-354.
- Sacerdoti, E. D. (1977). *A structure for plans and behavior*. Stanford Research Institute, Elsevier North-Holland Inc.
- Salveson, M.E. (1955). The assembly line balancing problem. *J. Ind. Engng.*, **6** (3), 18-25.
- Scholl, A. (1999). *Balancing and sequencing of assembly lines*. Heidelberg Physica.
- Shtub, A. and E.M. Dar El (1990). An assembly chart oriented assembly line balancing approach. *Int. J. of Prod. Res.*, **28** (6), 1137-1151.
- Sotskov, Yu. N. and A. Dolgui (2001). Stability Radius of the Optimal Assembly Line Balance with Fixed Cycle Time. In: *Proceedings of the IEEE Conference ETFA'2001*, pp. 623-628. IEEE Press.
- Süer, G.A. (1998). Designing parallel assembly lines. *Computers Ind. Eng.*, **35** (3-4), 467-470.
- Suresh, G. and S. Sahu (1994). Stochastic assembly line balancing using simulated annealing. *Int. J. Prod. Res.*, **32** (8), 1801-1810.
- Suresh, G., V. V. Vinod and S. Sahu (1996). A genetic algorithm for assembly line balancing. *Production Planning and Control*, **7**, 38-46.
- Sysoev, V. and A. Dolgui (1999). A Pareto optimization approach for manufacturing system design. In: *Proceedings of the International Conference on Industrial Engineering and Production Management (IEPM'99)*, Glasgow, Book 2, pp. 116-125. FUCAM.
- Talbot, F.B., J.H. Patterson, and W.V. Gehrlein (1986). A comparative evaluation of heuristic line balancing techniques. *Management Science*, **32**, 430-454.
- Tsai, D.M. and M.J. Yao (1993). A line-balanced base capacity planning procedure for series-type robotic assembly line. *Int. J. of Prod. Res.*, **31**, 1901-1920.
- Wee, T.S. and M.J. Magazine (1982). Assembly line balancing as generalized bin packing. *Operations Research Letters*, **1**, 56-58.